

Sistem Pakar Diagnosa Kerusakan Mesin Sepeda Motor *Matic* Berbasis Web Menggunakan *Certainty Factor*

Dedi Saputra¹, Deasy Purwaningtias², Windi Irmayani³

^{1,2}Program Studi Sistem Informasi, Fakultas Teknologi Informasi, UBSI Pontianak

³Program Studi Sistem Informasi Akuntansi, Fakultas Teknologi Informasi, UBSI Pontianak
dedi.dst@bsi.ac.id

Abstrak - Membuat keputusan memainkan peranan penting dalam proses diagnosa kerusakan mesin sepeda motor. Umumnya diagnosa kerusakan secara efektif dan efisien ini hanya bisa dilakukan oleh pakar yang ahli pada bidangnya. Dengan ditemukannya kecerdasan buatan dalam dunia komputer membuat hal tersebut menjadi lebih mudah. Penggunaan kecerdasan buatan berbasis komputer dapat menjadi pemecahan masalah bagi pelaku pengambil keputusan untuk membuat keputusan berdasarkan kondisi yang tidak pernah dialami sebelumnya dan upaya untuk mendapatkan diagnosa tersebut. Dengan kombinasi mesin inferensi dan basis pengetahuan menjadikan lahirnya aplikasi diagnosa kerusakan ini. Aplikasi ini dilengkapi juga dengan fitur pengembangan basis pengetahuan secara bertahap dengan mengandalkan interpretasi pengetahuan pakar yang dituangkan dalam sistem tersebut membuat sistem tersebut dapat memberikan diagnosa dengan baik dan akurat. Dalam usaha untuk mencapai penelitian, aplikasi ini menggunakan metode *Certainty Factor* untuk memperoleh diagnosa dapat memproses kemunculan kondisi tidak terstruktur. Selain itu, penerapan metode ini pada basis aturan menghasilkan derajat keyakinan terhadap masing-masing diagnosa kerusakan.

Kata kunci : *Sistem Pakar, Diagnosa, Certainty Factor*

1. a Latar Belakang

Sepeda Motor merupakan kendaraan yang banyak digunakan oleh orang Indonesia. Dibandingkan dengan mobil, motor sendiri memiliki tingkat fleksibilitas lebih tinggi dilihat dari kemudahan penggunaan dan ukuran kendaraan. Dengan berkembangnya teknologi otomotif lahirlah berbagai macam jenis motor. Selain motor bebek dan motor *sport* kini banyak orang lebih membeli dan menggunakan motor *matic*. Hal ini didukung dengan data pada Juni 2016 oleh Asosiasi Industri Sepedamotor Indonesia (AISI) yang menyebutkan “*market share* skutik (*matic*) masih sangat jauh meninggalkan segmen lain, dengan penguasaan sebesar 69,31 persen”. Motor *matic* sendiri merupakan wujud perkembangan otomotif yang selalu berinovasi dengan teknologi-teknologi terbaru. Dibandingkan motor bebek dan motor *sport*, motor *matic* memiliki beberapa keunggulan seperti mudah mengoperasikannya serta fitur-fitur tambahan dengan teknologi-teknologi mutakhir yang tidak ada pada motor bebek dan motor *sport*. Teknologi-teknologi mutakhir inilah yang menjadi keunggulan motor *matic* merebut perhatian para pengguna sepeda motor.

Dikarenakan perbedaan teknologi baik dari mesin maupun fitur-fitur yang tertanam dalam motor *matic*, banyak pengguna yang terbiasa menggunakan sepeda motor bebek merasa kebingungan ketika mesin motor *matic* mereka mengalami kerusakan. Akhirnya pengendara motor *matic* menyerahkan

sepenuhnya kepada mekanik untuk dilakukan pemeriksaan. Pengendara sepeda motor *matic* menjadi cenderung tidak peduli apakah kerusakan tersebut hanya diakibatkan oleh masalah yang sederhana atau rumit untuk diperbaiki.

Umumnya sebelum terjadi kerusakan pada sepeda motor *matic* yang digunakannya, pengendara akan merasakan gejala yang tidak biasa pada sepeda motor tersebut. Pengendara merasakan kejanggan atau ketidaknyamanan pada sepeda motor yang dikendarainya. Jika pengendara dapat memprediksi kerusakan yang terjadi berdasarkan gejala yang sudah ada, maka tentu kerusakan dapat dihindari sesegera mungkin dan dilakukan perbaikan. Pemasalahannya tidak semua orang memiliki pengetahuan tentang mesin sepeda motor *matic* layaknya para mekanik.

Sistem pakar yang pengusul buat adalah solusi untuk permasalahan yang ada, dimana sistem ini mampu mendiagnosa kerusakan mesin pada sepeda motor khusus berjenis *matic*. Sistem ini merupakan representasi pengetahuan beberapa mekanik yang tersertifikasi dan dituangkan dalam bentuk perangkat lunak sehingga dapat digunakan oleh para pengendara untuk memprediksi kerusakan yang akan terjadi.

Diharapkan dengan sistem pakar diagnosa kerusakan mesin sepeda motor *matic* ini dapat membantu para pengendara mendiagnosa kerusakan yang terjadi pada sepeda motor *matic* mereka sehingga dapat dilakukan perbaikan segera mungkin. Sistem

ini juga menambah pengetahuan sehingga dapat membantu pengendara untuk lebih mengerti mengenai mesin sepeda motor *matic* mereka.

1. b Rumusan

Masalah

Pada penelitian ini dapat dirumuskan masalah dalam penelitian ini adalah:

“Apakah sistem pakar diagnosa kerusakan mesin sepeda motor *matic* berbasis *web* dengan metode *certainty factor* dapat dilakukan?”

1. c Batasan Masalah

Ruang lingkup penelitian ini adalah pembuatan sistem pakar diagnosa kerusakan mesin sepeda motor *matic* berbasis *web* dengan metode *certainty factor*.

1. d Tujuan

Terdapat 2 tujuan dari penelitian ini, antara lain:

- a. Memberikan alternatif bagi masyarakat umum dalam melakukan diagnosis kerusakan mesin motor *matic*
- b. Membangun sistem pakar yang menyediakan informasi pada para pengendara sepeda motor yang belum berpengalaman sehingga dapat mengetahui gejala-gejala kerusakan yang akan terjadi pada sepeda motor *matic*.

1. e Manfaat Penelitian

- a. Sistem pakar ini diharapkan dapat membantu pengendara sepeda motor *matic* mendiagnosa kerusakan apa yang dialami sepeda motornya.
- b. Pengguna sistem pakar ini mendapatkan tambahan informasi mengenai sepeda motor *matic* terutama yang berhubungan dengan gejala dan diagnosa kerusakan pada motornya.
- c. Penelitian ini dapat dipublikasi pada jurnal ilmiah nasional ber ISSN dari hasil ini diharapkan dapat memberikan kontribusi dalam bidang otomotif terutama jenis sepeda motor *matic* mengenai diagnosa kerusakan pada mesin motor *matic*.

1. f Metode Penelitian

a. Lokasi dan Waktu Penelitian

Penelitian ini dilakukan pada unit kerja bengkel resmi Yamaha di Pontianak. Waktu penelitian selama 40 minggu. Pengumpulan data selama 8 minggu, analisis data selama 4 minggu, model perancangan selama 8 minggu, melakukan *coding* 16 minggu, implementasi selama 4 minggu.

b. Jenis dan Sumber Data

Jenis data yang digunakan pada penelitian ini berupa data primer dan data

sekunder dengan tujuan untuk mendapatkan data yang akurat. Data primer didapat dari wawancara teknisi yang telah disertifikasi oleh Yamaha dan data sekunder didapat dari arsip-arsip resmi dari Yamaha.

c. Teknik Pengumpulan Data

Data sekunder yang digunakan dalam penelitian ini diperoleh dengan menggunakan teknik pengumpulan data sebagai berikut:

1. Observasi
Data-data diperoleh melalui arsip-arsip resmi tentang sistem pakar yang diperoleh dari internet. Selain dari internet data-data juga diperoleh dari data-data statistik berupa jurnal atau buku yang telah dipublikasikan.
2. Wawancara
Mewawancarai para teknisi motor *matic* yang telah disertifikasi oleh Yamaha.
3. Studi Pustaka
Data-data diperoleh melalui arsip-arsip resmi tentang sistem pakar yang diperoleh dari para ahli pada bengkel resmi Yamaha. Selain itu data-data juga diperoleh dari data-data berupa jurnal atau buku yang telah dipublikasikan. Penulis melakukan penghimpunan informasi yang relevan dengan penelitian yang dilakukan oleh penulis. Setelah memperoleh data-data melalui studi pustaka penulis melakukan kajian terhadap fakta-fakta melalui buku-buku sepeda motor maupun artikel-artikel seputar mesin sepeda motor.

d. Model Pengembangan Perangkat Lunak

Metode pengembangan perangkat lunak yang digunakan dalam penelitian menggunakan metode air terjun (*waterfall*). Tujuan penggunaan metode tersebut bertujuan menjaga waktu pengerjaan perangkat lunak dengan waktu sesingkat mungkin tanpa melupakan kualitas.

Beberapa tahapan dalam metode air terjun (*waterfall*) yaitu sebagai berikut :

1. *Communication (Project initiation&Requirements)*
Tahapan awal dari dimulainya metode air terjun (*waterfall*). Tahapan ini dipergunakan untuk memperoleh fakta-fakta yang ada. Data-data yang telah diperoleh kemudian dipelajari dan dianalisa untuk dijadikan sumber penelitian.
2. *Planning (Estimating, scheduling and tracking)*
Data-data penelitian yang telah diperoleh sebelumnya direpresentasikan dalam bentuk pengetahuan. Kebutuhan

3. *Modeling (Analysis and design)*
Tahapan ini yaitu mengubah kebutuhan perangkat lunak menjadi sebuah model rancangan yang akan dipergunakan untuk generasi kode.
4. *Construction (Code and test)*
Dalam tahapan konstruksi ini, dilakukan generase kode melalui bahasa pemograman PHP, *Hyper Text Markup Language* (HTML), *Javascript* dan *Cascading Style Sheet* (CSS) . Kode-kode ini akan membentuk bagian-bagian program yang saling terhubung sesuai dengan fitur dan fungsi perangkat yang telah didefinisikan pada tahapan sebelumnya.
5. *Deployment (Delivery, support and feedback)*
Setelah tahapan konstruksi, data-data hasil penelitian dalam bentuk pengetahuan direpresentasikan dalam bentuk sistem secara bertahap, dimana setiap tahapannya dilakukan uji kasus untuk mengetahui kesesuaian hasil secara fungsional.

2. a. Dasar Teori

a. Sistem Pakar

Feigenbaum dalam Rosnelly (2012) menyatakan bahwa sistem pakar adalah “sebuah program pintar (*intelligent computer program*) yang memanfaatkan pengetahuan (*knowledge*) dan prosedur inferensi (*inference procedur*) untuk memecahkan masalah yang cukup sulit hingga membutuhkan keahlian khusus dari manusia”. Menurut Kusrini (2008) “Sistem pakar adalah aplikasi berbasis komputer yang digunakan untuk menyelesaikan masalah sebagaimana yang dipikirkan oleh pakar”. Sedangkan menurut Annisa (2018) “Sistem pakar merupakan sistem komputer yang mampu menirukan penalaran seorang pakar dengan keahlian pada suatu pengetahuan tertentu”.

Representasi Pengetahuan

Menurut Kusrini (2008) “Representasi Pengetahuan merupakan metode yang digunakan untuk mengkodekan pengetahuan dalam sebuah sistem pakar”.

Menurut Kusrini (2008) adapun karakteristik dari metode representasi pengetahuan adalah :

- a. Harus bisa diprogram dengan bahasa pemograman atau dengan *shells* dan hasilnya disimpan dalam memori.

- b. Dirancang sedemikian sehingga isinya dapat digunakan untuk proses penalaran.
- c. Model representasi pengetahuan merupakan sebuah struktur data yang dapat dimanipulasi oleh mesin inferensi dan pencarian untuk aktivitas pencocokan pola.

Metode Inferensi

Menurut Kusrini (2008) “Inferensi adalah Proses untuk menghasilkan informasi dari fakta yang diketahui atau diasumsikan.”. Dalam buku yang sama Kusrini menyatakan bahwa “Inferensi adalah Konklusi logis (*logical conclusion*) atau implikasi berdasarkan informasi yang tersedia”.

Dari beberapa definisi tersebut, maka dapat disimpulkan bahwa inferensi adalah proses untuk menghasilkan informasi dari fakta yang ada.

Kusrini (2008) membagi dua metode inferensi yang penting dalam sistem pakar yaitu :

a. Runut Maju (*Forward Chaining*)

Runut maju berarti menggunakan himpunan aturan kondisi-aksi. Dalam metode ini, data digunakan untuk menentukan aturan mana yang akan dijalankan, kemudian aturan tersebut dijalankan. Mungkin proses menambahkan data ke memori kerja. Proses diulang sampai ditemukan suatu hasil (Kusrini:2008).

Metode inferensi runut maju cocok digunakan untuk menangani masalah pengendalian (*controlling*) dan peramalan (*prognosis*) (Kusrini 2008). Ingin diperoleh konklusi dari daftar konklusi yang ada berdasarkan premis-premis dalam aturan dan fakta yang diberikan oleh pengguna. Berikut ini adalah daftar aturannya :

Aturan 9 :

Jika Premis 1
Dan Premis 2
Dan Premis 3
Maka Konklusi 1

Aturan 10 :

Jika Premis 1
Dan Premis 3
Dan Premis 4
Maka Konklusi 2

Aturan 11 :

Jika Premis 2
Dan Premis 3
Dan Premis 5
Maka Konklusi 3

Aturan 12 :

Jika Premis 1
Dan Premis 4
Dan Premis 5

Dan Premis 6
Maka Konklusi 4

Penelusuran maju pada kasus ini adalah untuk mengetahui apakah suatu fakta yang dialami oleh pengguna itu termasuk konklusi, konklusi 2, konklusi 3 atau konklusi 4 atau bahkan bukan salah satu dari konklusi tersebut, yang artinya sistem belum mampu mengambil keputusan karena keterbatasan aturan. Berdasarkan premis yang dipilih, maka sistem akan mencari aturan yang sesuai, sehingga akan diperoleh konklusinya.

Seandainya pengguna memilih Premis1, Premis2, dan Premis 3 maka aturan yang terpilih adalah aturan 1 dengan konklusinya adalah Konklusi 1. Seandainya pengguna memilih Premis 1 dan Premis 6 maka sistem akan mengarah pada aturan 12 dengan konklusi 4, tetapi karena aturan tersebut premisnya adalah Premis 1, Premis 4, Premis 5 dan Premis 6 maka premis-premis yang dipilih oleh pengguna tidak cukup untuk mengambil kesimpulan konklusi 4 sebagai konklusi terpilih.

b. Runut Balik (*Backward Chaining*)

Runut balik merupakan proses penalaran kebalikan dari runut maju. Dalam runut balik penalaran dimulai dengan tujuan kemudian menurut balik ke jalur yang akan mengarahkan ke tujuan tersebut. Runut balik juga disebut sebagai *goal-driven reasoning*, merupakan carar yang efisien dalam memecahkan masalah yang dimodelkan sebagai masalah pemilihan terstruktur.

Tujuan inferensi adalah mengambil pilihan terbaik dari banyak kemungkinan. Metode inferensi runut balik ini cocok digunakan untuk memecahkan masalah diagnosis.

Dengan menggunakan kasus yang sama pada proses penalaran runut maju, yang ingin didapatkan apda penalaran ini juga sama yaitu salah satu konklusi dari Konklusi 1, Konklusi 2, Konklusi 3, Konklusi 4 atau bahkan tidak dari keempat konklusi tersebut. Penelusuran didasarkan pada suatu keyakinan bahwa ada kemungkinan konklusi dari daftar konklusi merupakan salah satu tujuan atau konklusi terpilih berdasarkan fakta yang diberikan oleh pengguna. Sistem dengan urutan tertentu akan mengambil sebuah konklusi sebagai calon konklusinya. Misal urutannya adalah sesuai dengan urutan konklusi. Awalnya sistem akan mengambil hipotesis bahwa konklusinya adalah konklusi 1. Untuk membuktikan konklusinya, sistem akan mencari premis-premis aturan yang mengandung konklusi 1. Setelah itu sistem akan meminta umpan balik kepada pengguna

mengenai premis-premis yang ditemukan tersebut.

Faktor Kepastian

Menurut *Giarattano* dan *Riley* dalam *Kusrini* (2008:15) menyatakan bahwa "*Certainty Factor* (CF) merupakan nilai parameter klinis yang diberikan MYCIN untuk menunjukkan besarnya kepercayaan". Sedangkan menurut *Rosnelly*(2012:89) menyatakan bahwa "*Certainty Factor* (CF) menunjukkan ukuran kepastian terhadap suatu fakta atau aturan".

Dari beberapa definisi tersebut, maka dapat disimpulkan bahwa faktor kepastian adalah nilai parameter yang menunjukkan ukuran kepastian berdasarkan ukuran terhadap suatu fakta dan aturan.

Menurut *Giarattano* dan *Riley* dalam *Kusrini* (2008) *Certainty Factor* didefinisikan sebagai berikut :

$$CF(H,E) = MB(H,E) - MD(H,E)$$

Penjelasan :

CF(H,E) : *Certainty Factor* dari hipotesis H yang dipengaruhi oleh gejala (*evidence*) E.

Besarnya CF berkisar antara -1 sampai dengan 1. Nilai -1 menunjukkan ketidakpercayaan mutlak, sedangkan nilai 1 menunjukkan kepercayaan mutlak.

MB(H,E) : Ukuran kenaikan kepercayaan (*measure of increased belief*) terhadap hipotesis H yang dipengaruhi oleh gejala E.

MD (H,E) : Ukuran kenaikan ketidakpercayaan (*measure of increased disbelief*) terhadap hipotesis H yang dipengaruhi oleh gejala E.

CF Gabungan merupakan akhir dari sebuah calon konklusi. CF ini dipengaruhi oleh semua CF paralel dari aturan yang menghasilkan konklusi tersebut. CF Gabungan diperlukan jika suatu konklusi diperoleh dari beberapa aturan sekaligus. CF akhir dari satu aturan dengan aturan yang lain digabungkan untuk mendapatkan nilai CF akhir bagi calon konklusi tersebut. Berikut adalah rumus CF Gabungan yaitu

$$CF(x,y) = \begin{cases} CF(x) + CF(y) - (CF(x) \times CF(y)), & \text{CF}(x) > 0 \text{ dan } CF(y) > 0 \\ \frac{CF(x) + CF(y)}{(1 - (\min(|CF(x)|, |CF(y)|)))}, & \text{salah satu } (CF(x), CF(y)) < 0 \\ CF(x) + (CF(y) \times (1 + CF(x))), & \text{CF}(x) > 0 \text{ dan } CF(y) < 0 \end{cases}$$

Sumber : <http://www.metode-algoritma.com/2013/06/aplikasi-system-pakar->

menggunakan-metode.html (2016)

Gambar 1 Rumus CF Gabungan

2. b. Kajian Pustaka

Berikut ini merupakan tinjauan dari hasil penelitian terdahulu yang berkaitan dengan penelitian Sistem Pakar Diagnosa Kerusakan Mesin Sepeda Motor *Matic* Berbasis *Web* Menggunakan *Certainty Factor*

a. Penelitian yang dilakukan oleh Imam Wicaksono, Fitro Nur Hakim dan Victor Gayuh Utomo dengan judul Sistem Pakar Diagnosa Kerusakan Pada Motor *Matic* Vario Berbasis *Web* pada Jurnal Transformatika. Volume 13 Nomor 2, Januari 2016.

Penelitian ini menggunakan data kerusakan, gejala, penyebab dan solusi dari kerusakan motor *matic* vario. Penelitian ini membuat sistem pakar dengan web yang meliputi menu kerusakan, solusi, gejala, aturan dan logout.

b. Penelitian yang dilakukan oleh Dian Kusuma Wati dan Wiwin Kuswinardi dengan judul Sistem Pakar Diagnosa Kerusakan Sepeda Motor *Matic* Injeksi Menggunakan Metode *Dempster Shafer* pada Jurnal Bimasakti Volume 1 Nomor 3 . April, 2015.

Penelitian ini menggunakan metode *dempster shafer* yang merupakan teori yang digunakan untuk mendefinisikan pemikiran secara luas dai beberapa pendekatan umum yang serupa, untuk beberapa jenis dari situasi. Penelitian ini belum dibuat secara *mobile* dan terbatas pada motor *matic* injeksi.

c. Penelitian yang dilakukan Cholil Jamhari, Agus Kiryanto dan Sri Huning Anwariningsih berjudul Sistem Pakar Diagnosis Kerusakan Sepeda Motor *Non Matic*, disampaikan pada Seminar Nasional IENACO.

Penelitian ini bertujuan untuk merancang sistem pakar dibidang otomotif khususnya sepeda motor *non matic*. Perencanaan sistem dalam membuat *knowledge base* memakai pohon keputusan dan kaidah produksi sebagai representasi pengetahuan. Pembuatan metode inferensi memakai metode *forward chaining* yang telah dimodifikasi sehingga sesuai dengan permasalahan.

3. Analisa dan Perancangan Sistem

a. Identifikasi Kebutuhan

Identifikasi dibedakan menjadi dua bagian sebagai berikut :

1. Kebutuhan Fungsional

Kebutuhan fungsional merupakan kebutuhan utama yang diharapkan sistem yang berkaitan langsung dengan sistem yang dibuat. Adapun kebutuhan fungsional yang dibutuhkan agar sistem dapat melakukan pengolahan data-data masukkan dan menghasilkan keluaran yang dibutuhkan adalah antara lain :

a. Prosedur *login*

Merupakan suatu proses yang harus dilakukan pada awal menggunakan aplikasi oleh *administrator*. Prosedur *login* dapat dilakukan dengan memasukkan *username* dan *password*. Jika *username* dan *password* cocok maka sistem akan mengarahkan *administrator* dari halaman *login* ke halaman *review* pengunjung.

b. Diagnosa Kerusakan

Sebuah fitur diagnosa penyakit dimana pengguna dapat melihat keluaran berdasarkan gejala yang telah dipilih secara simultan. Data gejala dan penyakit yang digunakan dalam diagnosa ini berdasarkan pada pakar yang telah dipilih oleh pengunjung. Dari gejala-gejala yang telah dimasukkan oleh pengunjung maka sistem akan melakukan proses penalaran dan kemudian akan menyajikan semua kemungkinan yang mungkin terjadi berdasarkan gejala-gejala yang ada. Data keluaran tersebut diurutkan berdasarkan ranking.

c. Statistik dan *review* pengunjung

Fitur ini merupakan sebuah umpan balik kepada para *administrator*. Dikatakan sebagai umpan balik karena setiap pengunjung melakukan diagnosa penyakit maka pada akhir diagnosa pengunjung dapat memberikan umpan balik beserta *rating* terhadap *administrator* tersebut. Sehingga *administrator* dapat melakukan perbaikan terutama pada pemutakhiran data-data basis pengetahuan untuk membuat diagnosa tersebut menjadi lebih tepat.

d. Fasilitas Penjelasan

Fasilitas penjelasan merupakan sebuah fitur yang dibutuhkan untuk mendukung pengambilan keputusan. Fasilitas penjelasan ini didasarkan pada gejala-gejala yang diterima oleh sistem

menjelaskan bagaimana sebuah keputusan diperoleh dan juga menampilkan nilai kemungkinan dari suatu konklusi melalui metode *Certainty Factor* (CF). Nilai CF dikalkulasikan berdasarkan jawaban pengguna dengan nilai CF pada setiap aturan yang telah ada sebelumnya melalui rumus persamaan yang telah diprogramkan sebelumnya.

- e. **Pengelolaan basis pengetahuan**
Basis pengetahuan merupakan hal yang menjadi tumpuan pada aplikasi ini. Data-data basis pengetahuan digunakan sebagai acuan pada saat diagnosa dijalankan. Pada prosedur ini terdapat fitur penambahan, penyuntingan, penghapusan serta konfigurasi aturan dari data-data yang akan digunakan. Data-data yang digunakan dalam prosedur ini terdiri dari data penyakit, data detail-penyakit, data gejala dan data aturan. Data-data tersebut dimasukkan oleh pengguna akan divalidasi dahulu oleh program. Jika telah benar maka program akan melakukan *entry* data tersebut pada *database* sesuai dengan tipe masing-masing. Apabila validasi gagal maka aplikasi akan memberitahu kepada pengguna untuk melakukan perbaikan pada masukan tersebut.

2. Kebutuhan Non Fungsional

Kebutuhan non fungsional meliputi kebutuhan penunjang atau kebutuhan yang menjadi pendukung dari sistem yang dibuat, adapun kebutuhan tersebut antara lain :

- a. **Kebutuhan Perangkat Keras**
Agar aplikasi ini dapat berjalan dengan baik maka dibutuhkan minimal spesifikasi perangkat keras untuk menunjang kinerja sistem. Adapun kebutuhan *hardware* yang diperlukan adalah sebagai berikut :
 - 1.) CPU (*Central Processing Unit*)
 - a.) Processor : 1.4Ghz
 - b.) RAM: 2 GB DDR3
 - c.) Harddisk : 20 GB
 - 2.) Monitor (14" WXGA LCD / LED 1028x768)
 - 3.) Keyboard (*Standard Keyboard* 86 keys)
 - 4.) Mouse
- b. **Kebutuhan Perangkat Lunak**
Adapun kebutuhan perangkat lunak yang diperlukan agar perangkat lunak dapat berjalan yaitu :
 - 1.) Sistem Operasi : Windows 8.1
 - 2.) Web server : XAMPP 3.2.1

- 3.) Web Browser : Mozilla Firefox atau Google Chrome
- 4.) DBMS : MySQL
- c. Kebutuhan *Brainware*

Brainware atau pengguna dalam aplikasi ini dapat dibedakan menjadi 2 pengguna yaitu : Pengguna Umum dan *Administrator*. Pengguna umum didefinisikan sebagai pengguna yang dapat menggunakan aplikasi tanpa melalui prosedur *login*. Berbeda dengan *administrator* yang harus melewati prosedur *login*. Pembagian kategori *Brainware* dibagi berdasarkan hak aksesnya sebagai berikut

1.) Pengguna umum

Pengguna umum dapat memanfaatkan fitur-fitur yang disediakan pada aplikasi ini seperti melakukan diagnosa kerusakan dengan melakukan interaksi antarmuka program. Setelah melakukan diagnosa, pengguna umum dapat melihat hasil kerusakan yang telah disimpulkan terhadap gejala-gejala yang dipilih. Setelah selesai pengguna dapat mengulangi kembali dari awal untuk melakukan diagnosa kerusakan.

2.) Administrator

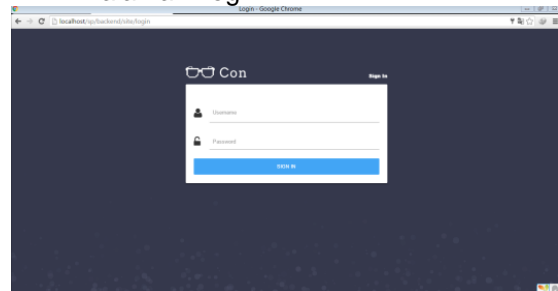
Administrator merupakan pengguna yang dibutuhkan untuk menjalankan aplikasi ini setelah melewati proses *login*. Fitur-fitur yang disediakan dalam *administrator* seperti melakukan pengolahan data penyakit, gejala, aturan (*rule CF*), serta melihat statistik pengunjung dan *review* yang diberikan pengunjung guna sebagai perbaikan untuk *administrator* kedepan.

3. Implementasi Sistem dan Hasil

a. Implementasi Rancangan Antar Muka Administrator.

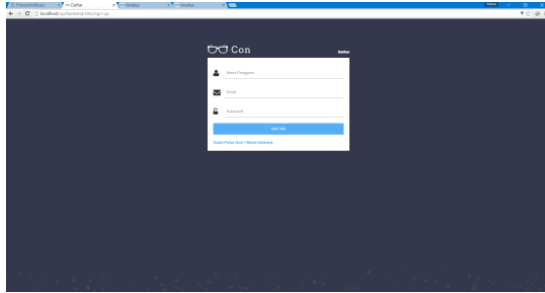
Berbeda dengan pengunjung yang tidak memerlukan prosedur *login*. Halaman *administrator* memerlukan prosedur *login* untuk mengakses halaman tersebut. Berikut adalah implementasi rancangan antar muka *administrator* :

1. Implementasi Rancangan Antar Muka Halaman Login



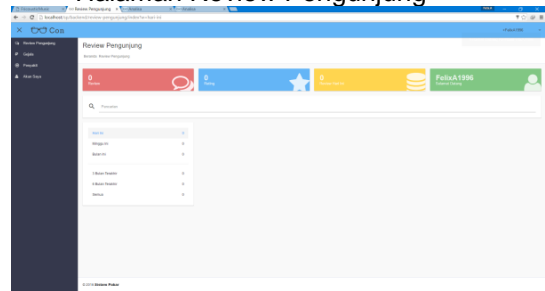
Gambar 2 Implementasi Rancangan Antar Muka Halaman Login

2. Implementasi Rancangan Antar Muka Halaman Daftar



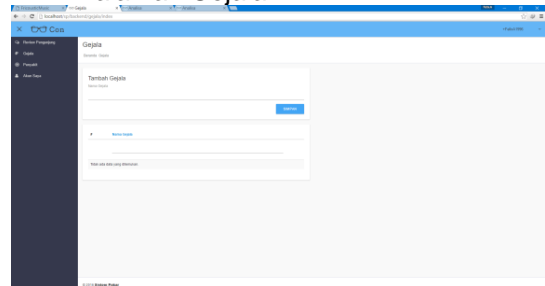
Gambar 3 Implementasi Rancangan Antar Muka Halaman Daftar

3. Implementasi Rancangan Antar Muka Halaman Review Pengunjung



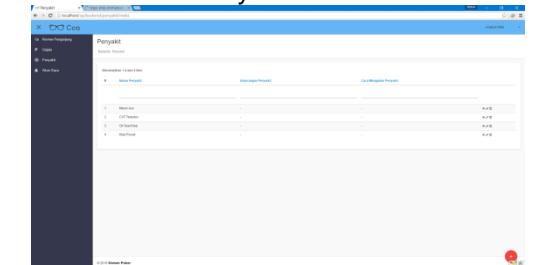
Gambar 4 Implementasi Rancangan Antar Muka Halaman Review Pengunjung

4. Implementasi Rancangan Antar Muka Halaman Gejala



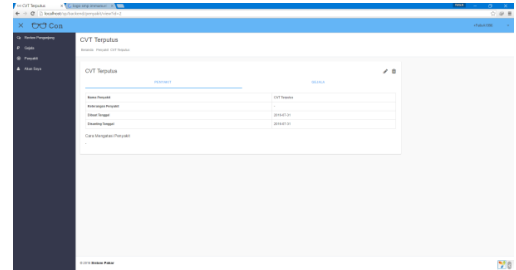
Gambar 5 Implementasi Rancangan Antar Muka Halaman Gejala

5. Implementasi Rancangan Antar Muka Halaman Penyakit

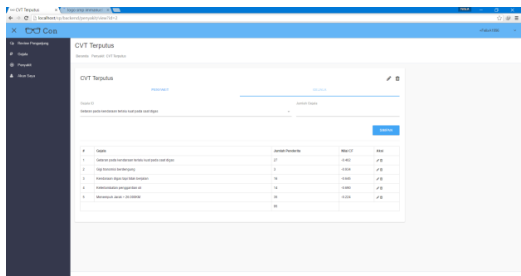


Gambar 6 Implementasi Rancangan Antar Muka Halaman Penyakit

6. Implementasi Rancangan Antar Muka Halaman Detail Penyakit

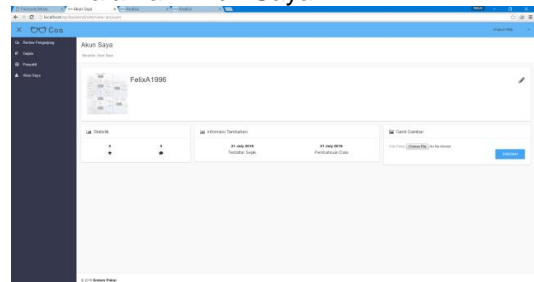


Gambar 7 Implementasi Rancangan Antar Muka Halaman Detail Penyakit Tab Penyakit



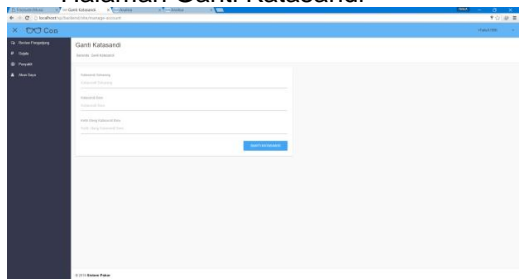
Gambar 8 Implementasi Rancangan Antar Muka Halaman Detail Penyakit Tab Gejala

7. Implementasi Rancangan Antar Muka Halaman Akun Saya



Gambar 9 Implementasi Rancangan Antar Muka Halaman Akun Saya

8. Implementasi Rancangan Antar Muka Halaman Ganti Katasandi



Gambar 10 Implementasi Rancangan Antar Muka Halaman Ganti Katasandi

5. Penutup Kesimpulan

1. Dalam penelitian ini menghasilkan perangkat lunak yang dipergunakan untuk melakukan diagnosa kerusakan pada mesin sepeda motor *matic* Yamaha *Mio J Sporty* menggunakan

- XAMPP sebagai *web server*, MySQL sebagai basis data, PHP sebagai bahasa pemrograman.
2. Kelengkapan basis pengetahuan yang ditanamkan dalam aplikasi sangat mempengaruhi hasil diagnosa yang ada. Serta penerapan metode *Certainty Factor* pada aplikasi ini membantu memberikan penjelasan terhadap hasil diagnosa berdasarkan derajat keyakinan dari setiap aturan yang ada jika diberikan gejala-gejala tertentu.
 3. Aplikasi ini digunakan untuk melakukan diagnosa kerusakan berdasarkan basis pengetahuan yang telah ditanamkan serta tidak dapat memberikan penalaran diluar batas data yang telah ditanamkan sebelumnya.

DAFTAR PUSTAKA

- [1] Annisa, Riski. (2018). Sistem Pakar Metode *Certainty Factor* Untuk Mendiagnosa Tipe Skizofrenia. Indonesian Journal on Computer and Information Technology, 3, 40-46.
- [2] Jamhari, Cholil, dkk. (2014). Sistem Pakar Diagnosis Kerusakan Sepeda Motor Non *Matic*. Seminar Nasional IENACO, 374-382.
- [3] Kusrini. (2008). Aplikasi Sistem Pakar Menentukan Faktor Kepastian Pengguna dengan Metode Kuantifikasi Pertanyaan. Yogyakarta: Andi Offset
- [5] Rosnelly, Rika. (2012). Sistem Pakar Konsep dan Teori. Yogyakarta: Andi Offset.
- [6] Wati, Dian Kusuma, dkk. 2015. Sistem Pakar Diagnosa Kerusakan Sepeda Motor Matic Injeksi Menggunakan Metode Dempster Shafer. BIMASAKTI, 1, 1-4.
- [7] Wicaksono, Imam, dkk. 2016. Sistem Pakar Diagnosa Kerusakan Pada Motor Vario Berbasis Web. Jurnal Transformatika, 3, 49-58.